

I claim:

1. A method of compressing a document, comprising:
preparing an encoded representation of a document by scanning the document to
provide a scanner output;
5 classifying the scanner output as belonging to a class of document taken from the
document classes consisting of smooth, text, graphics and image; and
adaptively compressing the scanner output as a function of the class of the
document.
- 10 2. The method of claim 1 wherein said classifying includes setting an order of
priority to the classification of smooth > text > graphics > image.

3. The method of claim 1 wherein said classifying includes:
generating a histogram of the scanner output and identifying maximum and
minimum values for the scanner output;
selecting contrast threshold values to identify a text class document and a smooth
5 class document; and
using a sequential classification scheme:
classifying a document as smooth when the difference between the maximum and
minimum histogram value is below the contrast threshold value for smooth class documents;
classifying a document as text when $|c_1 - c_2| > 128$ and $p >$ the contrast threshold
10 value for text, wherein two dominant values, c_1, c_2 , are found and the cumulative probability, p is
the summation over $[c_1 - A, c_1 + A]$ and $[c_2 - A, c_2 + A]$);
classifying a document as graphic when the number of dominant values n belongs
to $[1, M]$ and the cumulative probability p is the summation over all $[c_k - A, c_k + A]$ for $k=1, \dots, n) >$
the contrast threshold value for graphics;
15 otherwise, classifying the document as an image for any scanner output not
already classified.

4. The method of claim 1 wherein said adaptively compressing includes compressing
the scanner output using a compression scheme taken from the group of compression schemes
20 including, singular mode, binary mode, M-ary mode and continuous mode.

5. The method of claim 4 which further includes compressing smooth class scanner output using singular mode compression; compressing text class scanner output using binary mode compression; compressing graphic class scanner output using M-ary compression mode and compressing image class scanner output using continuous mode compression.

5

6. The method of claim 4 which further includes post processing text class scanner output using a low-pass filter.

7. A method of compressing a document, comprising:
preparing an encoded representation of a document by scanning the document to
provide a scanner output;

5 document classes consisting of smooth, text, graphics and image, including:

generating a histogram of the scanner output and identifying maximum
and minimum values for the scanner output;

selecting contrast threshold values to identify a text class document and a
smooth class document; and

using a sequential classification scheme:

classifying a document as smooth when the difference between the
maximum and minimum histogram value is below the contrast threshold value for smooth class
documents;

classifying a document as text when $|c_1 - c_2| > 128$ and $p >$ the contrast
threshold value for text, wherein two dominant values, c_1, c_2 , are found and the cumulative
probability, p is the summation over $[c_1 - A, c_1 + A]$ and $[c_2 - A, c_2 + A]$);

classifying a document as graphic when the number of dominant values n
belongs to $[1, M]$ and the cumulative probability p is the summation over all $[c_k - A, c_k + A]$ for
 $k=1, \dots, n) >$ the contrast threshold value for graphics;

otherwise, classifying the document as an image for any scanner output not
already classified.

; and

adaptively compressing the scanner output as a function of the class of the document, including compressing the scanner output using a compression scheme taken from the group of compression schemes including, singular mode, binary mode, M-ary mode and continuous mode.

5

8. The method of claim 7 which further includes compressing smooth class scanner output using singular mode compression; compressing text class scanner output using binary mode compression; compressing graphic class scanner output using M-ary compression mode and compressing image class scanner output using continuous mode compression.

9. The method of claim 8 which further includes post processing text class scanner output using a low-pass filter.

10. A compression apparatus for compressing scanned data, comprising:
a scanner for scanning a document and generating a scanner output;
a block-based classifier for classifying the scanner output as belonging to a class
of documents taken from the document classes consisting of smooth, text, graphics and image;
5 an adaptive compressor for compressing the scanner output according to a
compression mode as a function of the class of document;
a storage mechanism for storing compressed scanner output and compression
mode information; and
a decompressor for decompressing compressed scanner output in accordance with
10 the compression mode information.

11. The apparatus of claim 10 wherein said block-based classifier includes an order of
priority for the classification of smooth > text > graphics > image.

12. The apparatus of claim 10 wherein said block-based classifier includes a
classification mechanism for classifying a document, including a histogram generator for
generating a histogram of scanner output, and a threshold selecting mechanism for selecting
contrast threshold values associated with at least two classes of documents.

13. The apparatus of claim 10 wherein said adaptive compressor includes compression algorithms, including a singular mode compression algorithm, a binary mode compression algorithm, a M-ary mode compression algorithm and a continuous mode compression algorithm; and wherein smooth class scanner output is compressed using singular mode compression; text class scanner output is compressed using binary mode compression; graphic class scanner output is compressed using M-ary compression mode and image class scanner output is compressed using continuous mode compression.

14. The apparatus of claim 13 which further includes a low-pass filter for use with text class scanner output during post-processing.